APPLICATION FOR UNITED STATES PATENT

SYSTEMS AND METHODS FOR DIRECT NAVIGATION TO SPECIFIC PORTION OF TARGET DOCUMENT

INVENTORS: **David P. Marmaros**
450 Del Medio
Mountain View, CA 94040
A Citizen of Canada

Benjamin Gomes
79 Eldora Drive
Mountain View, CA 94041
A Citizen of United States

Krishna Bharat
1458 Sierra Creek Way
San Jose, CA 95132
A Citizen of India

ASSIGNEE: **Google Inc.**
2400 Bayshore Parkway
Mountain View, CA 94043
A DELAWARE CORPORATION

ENTITY: **Large**

Jung-hua Kuo
Attorney at Law
P.O. Box 3275
Los Altos, CA 94024
Tel: (650) 988-8070
Fax: (650) 988-8090

SYSTEMS AND METHODS FOR DIRECT NAVIGATION TO SPECIFIC PORTION OF TARGET DOCUMENT

BACKGROUND OF THE INVENTION

5 ### 1. Field of the Invention

[0001]   Systems and methods for direct navigation to and/or highlighting a specific portion of a target document such as query-relevant portion of the document are disclosed.

10 ### 2. Description of Related Art

[0002]   When a user searches for information on the Internet by submitting a query to a search engine, the search engine returns a results page that often provides several hyperlinks to web pages that may be relevant to the user's query. Under each hyperlink, the results page typically also provides query-relevant information or text, often referred

15 to as "snippets," extracted from the webpage to which the corresponding hyperlink refers.

[0003]   When the user clicks on a desired hyperlink on the search results page, the client browser typically navigates to the top of the target or destination webpage (or other document such as a Word, Excel or Portable Document Format (PDF) document, for example) to which the hyperlink refers. However, the target webpage may be long and/or

20 the search query terms may be buried deep within the webpage. To locate the desired search terms, the user may perform a search for the desired text from within the web browser and/or manually (visually) search for the relevant information by scrolling through the webpage.

[0004]   As an example, to find pizza places in or near zip code 94043, the user may

25 search for "pizza" at zip code 94043 using Google's "Searching by Location" search engine currently located at http://labs.google.com/location (i.e., http://labs.google.com/location?q=pizza&geo_near=94043&Search=Google+Search) as shown in **FIG. 1**. An exemplary results page returned by the search engine is shown in **FIG. 2** in which the first hyperlink 20 points to http://www.waiter.com/roundtable/ which

30 contains a list of 35 pizza places in or near zip code 94043. In addition to the various hyperlinks, the search engine also provides a snippet 26 extracted from the webpage to

which each hyperlink refers. As shown, certain text of each snippet 26 is in bold font so as to highlight certain text that may be useful as determined by the search engine. For the first hyperlink 20, the search engine highlights the text "Pizza 570 N. Shoreline Blvd. MOUNTAIN VIEW, CA." of the snippet 26 in bold.

[0005]   If the user clicks on the first-listed hyperlink 20, the client browser loads and displays the destination webpage 22 to which the hyperlink 20 refers at the top of the webpage 22, as shown in **FIG. 3**. However, the target document or webpage is typically not hosted or authored by the same entity as that of the search engine or other source or origin document such that the desired information is typically not propagated from the source document to the target document. Thus, for example, because the browser typically displays the webpage 22 at the top of the webpage 22, the restaurant that is in or closest to zip code 94043 may or may not be initially displayed within the browser. In the current example, the restaurant that is in or closest to zip code 94043, namely, the Round Table Pizza at 570 N. Shoreline Blvd. in Mountain View, CA is the tenth out of 35 listings on the destination webpage 22.

[0006]   In order for the user to determine which of the listings is the most query-relevant listing, the user can determine as the user scrolls scroll through the 35 listings on the destination webpage 22 which of the listings is the one that is most relevant to the user's query, i.e., in or closest to zip code 94043. In the current example, the user must scroll down a number of screens before the query-relevant listing 24 is even displayed within the browser, as shown in **FIG. 4**. Alternatively, the user may take note of the content of the snippet 26 before navigating or clicking away from the results page (**FIG. 2**) and then search for (either visually by scrolling or by performing a text search from within the browser) the content of the snippet.

[0007]   As is evident, each result on the search results page generated by the search engine only points to a webpage and not to anything more granular. In other words, the snippet generated by the search engine is only displayed to the user on the search results page and is not propagated or otherwise utilized to directly navigate to the desired location on the destination webpage.

SUMMARY OF THE INVENTION

[0008]   Systems and methods for direct navigation to and/or highlighting a specific portion of a target document such as query-relevant portion of the document are disclosed. It should be appreciated that the present invention can be implemented in numerous ways, including as a process, an apparatus, a system, a device, a method, or a computer readable medium such as a computer readable storage medium or a computer network wherein program instructions are sent over optical or electronic communication lines. Several inventive embodiments of the present invention are described below.

[0009]   In one embodiment, the method may include generating a search result associated with a locator or link, e.g., a Universal Resource Identifier (URI) or a Uniform Resource Locator (URL), to a search result document in response to a search query from a client device, generating an instruction corresponding to the search result, the instruction being to a document browser on the client device to navigate directly to an intra-document portion related to the query within the search result document when the search result is selected by a user, and providing the search result to the user. The search result may include a snippet extracted from the search result document such that the instruction causes navigation directly to at least a portion of the snippet in the search result document or to a portion that the sever or search engine determined to be similar or otherwise relevant. The instruction may be an intra-document link, i.e., a link to a specific portion within the document, containing an artificial anchor that is undefined in the search result document, e.g., one designated by a preassigned artificial anchor designator. Alternatively, the instruction may be an intra-document link containing a named anchor defined in the search result document. As yet another alternative, the instruction may be a hidden tag or an attribute on a tag in a search result page. The intra-document portion may be text and/or image within the search result document, a tag identification, a tag name, a tag location in the document object model, an HTML byte offset defined in the document, and/or a general offset or location definition within the document.

[0010]   In yet another embodiment, a method for linking to an intra-document portion of a target document generally includes generating an artificial intra-document anchor, i.e., an anchor that is undefined in the search result document, corresponding to the intra-

document portion within the target document. The artificial anchor may be appended to a link pointing to the target document.

[0011]   In yet a further embodiment, a method for navigating directly to an intra-document portion within a target document generally includes receiving a document link to the target document and executing an instruction to navigate directly to and optionally highlighting the intra-document portion within the target document in response to the document link being selected, the instruction being one of a reference to an artificial anchor appended to the document link, the artificial anchor being undefined in the target document, a hidden tag defined in the target document, and an attribute on a tag defined in the target document.

[0012]   In yet another embodiment, a method generally includes receiving a search result page including a search result link pointing to a search result document and a snippet extracted from an intra-document portion within the search result document, generating an intra-document link based on the snippet by appending an artificial anchor to the search result link, the intra-document link pointing to the intra-document portion within the search result document and the artificial anchor being undefined in the target document, and navigating directly to the intra-document portion within the target document in response to selection of the intra-document link.

[0013]   In yet a further embodiment, a computer program product embodied on a computer-readable medium, the computer program product including instructions, which when executed by a computer system, are operable to cause the computer system to perform acts generally including generating at least one search result in response to receiving a search query from a user, the search result being associated with a search result document link to a search result document, generating an instruction corresponding to the search result, the instruction being to a document browser to display the search result on the client device, the instruction being to navigate directly to the intra-document portion within the search result document when the search result is selected by the user, the intra-document portion being related to the search query, and providing the search result to the user.

[0014]   These and other features and advantages of the present invention will be presented in more detail in the following detailed description and the accompanying figures which illustrate by way of example the principles of the invention.

BRIEF DESCRIPTION OF THE DRAWINGS

5   [0015]   The present invention will be readily understood by the following detailed description in conjunction with the accompanying drawings, wherein like reference numerals designate like structural elements.

[0016]   FIG. 1 is an illustrative client browser window displaying a search engine interface for entering a query.

10   [0017]   FIG. 2 is an illustrative client browser window displaying search results.

[0018]   FIG. 3 is an illustrative client browser window displaying a webpage to which a search result hyperlink refers at the top of the webpage.

[0019]   FIG. 4 is an illustrative client browser window displaying the webpage of FIG. 3 at a location that displays query-relevant information within the browser.

15   [0020]   FIG. 5 is an illustrative client browser window displaying the webpage of FIG. 3 in which the display is automatically scrolled to the query-relevant information of the webpage and the query-relevant information highlighted.

[0021]   FIG. 6 is a flowchart of an illustrative client-side process for automatically scrolling to and highlighting the query-relevant information of the webpage.

20   [0022]   FIG. 7 is a flowchart of an illustrative server-side process for generating and inserting an artificial named anchor in the URL of each search result.

[0023]   FIG. 8 is a flowchart of a modified portion of the process shown in FIG. 7 in which the search engine also determines if the target webpage already contains a named anchor at or near the snippet.

25   [0024]   FIG. 9 is a block diagram of an illustrative network system.

[0025]   FIG. 10 is a block diagram of an illustrative client device.

DESCRIPTION OF SPECIFIC EMBODIMENTS

[0026]   Systems and methods for direct navigation to and/or highlighting a specific portion of a target document such as query-relevant portion of the document are

disclosed. The following description is presented to enable any person skilled in the art to make and use the invention. Descriptions of specific embodiments and applications are provided only as examples and various modifications will be readily apparent to those skilled in the art. The general principles defined herein may be applied to other

5   embodiments and applications without departing from the spirit and scope of the invention. Thus, the present invention is to be accorded the widest scope encompassing numerous alternatives, modifications and equivalents consistent with the principles and features disclosed herein. For purpose of clarity, details relating to technical material that is known in the technical fields related to the invention have not been described in detail

10   so as not to unnecessarily obscure the present invention.

[0027]   As discussed above, **FIG. 1** is an illustrative client browser window displaying the interface for a Google search engine that searches by location. **FIG. 2** is an illustrative client browser window displaying the search results in response to a query for "pizza" and "94043." in which the first hyperlink 20 points to

15   http://www.waiter.com/roundtable/ that contains a list of 35 pizza restaurants. In addition to the various hyperlinks, the search engine also provides a snippet 26 extracted from the webpage to which each hyperlink refers. As shown, certain text of each snippet 26 is in bold font so as to highlight certain text that may be useful as determined by the search engine. As an example, for the first hyperlink 20, the search engine highlights the query-

20   relevant text "Pizza 570 N. Shoreline Blvd. MOUNTAIN VIEW, CA." of the snippet 26 in bold. Conventionally, the user may click on the first hyperlink 20 to cause the browser to navigate to the top of the webpage to which the hyperlink 20 refers, as shown in **FIG. 3**. The user may then scroll down to find and determine which of the 35 listed pizza restaurants is the one closest to or located in zip code 94043 within the webpage 22 as

25   shown in **FIG. 4**. Alternatively, the user may take note of the content of the snippet in the results page (**FIG. 2**) before navigating from the results page to the destination webpage and then search for (either by scrolling or by performing a text search from within the browser) the content of the snippet.

[0028]   Conventional named anchors are added by webpage authors to allow users to

30   navigate directly to a specific part or location within the webpage. For example, the webpage author may create a named anchor with <A NAME = location> Location </A>

at the specific part of the webpage to which the reader can navigate directly from another part of the same webpage or from a different webpage. The NAME attribute names a section, e.g., text or image, so that other links can reference it via a clickable HREF link containing a pound sign "#" followed by the anchor name. In particular, the same or

5 different webpage may include a clickable HREF link: <A href = "*[URL]*#location"> Jump to Location </A>, where *[URL]* is the URL of the target webpage. The HREF attribute specifies the URL address to which the browser is to navigate when the user clicks on the HREF link, i.e., "Jump to Location." Note that using the HREF link containing an anchor name requires that the author of the webpage to have created the

10 named anchor. If a named anchor does not exist for the specific part of the webpage, then the HREF link cannot be used to allow the reader to navigate directly to that specific part of the webpage. If an HREF link refers to a non-existent anchor, the browser simply navigates to the top of the target webpage.

[0029] With systems and methods described herein, mechanisms are provided to

15 generate or simulate links with artificial named anchors and to allow the browser to recognize the artificial named anchor and navigate directly to the desired specific part of the target webpage even when the author of the webpage has not created a named anchor at the specific part of the webpage. In particular, the systems and methods described herein simulate the general functionality of the named anchor and the HREF link to

20 provide links containing artificial named anchors that allow navigation directly to a specific part of the target webpage even when a named anchor does not exist at the specific part of the target webpage. Such links can be utilized by any webpage to provide a link to a specific part of another target webpage. In particular, such links containing artificial named anchors can be particularly useful for search result pages returned by

25 search engines.

[0030] The systems and methods described herein may be utilized to navigate any document that may be provided from any suitable source as the Internet, an intranet, or local memory, for example. The term URL as used herein generally refers to a locator or address for any document, not necessarily only those available on the Internet. Suitable

30 target or destination documents are typically those displayable by a client browser,

typically a client web browser, such as an Internet Explorer, Netscape, Opera, or Mozilla browser.

[0031]   Referring again to the example described above with reference to **FIGS. 1-4,** the search results page returned by the search engine may provide or otherwise simulate links to a specific part of a target webpage such as the part of the target webpage that includes at least a portion of the snippet 26 or to a portion that the sever or search engine determined to be similar or otherwise relevant. Thus when the user clicks on a portion of the snippet 26, for example, the browser may navigate directly to the part of the target webpage 22 that includes the portion of the snippet 26, as shown in **FIG. 5**.

[0032]   To further draw the user's attention to the query-relevant listing 24 in the target webpage 22, at least some of the snippet, e.g., "570 N. Shoreline," may be highlighted. For example, a specific portion may be highlighted, i.e., draw attention to or otherwise emphasize, by modifying the format of the specific portion such as by underlining, bolding, italicizing, foreground and/or background color changing, font and/or size changing, border drawing, text animating (e.g., "marching red ants", etc.), aligning, kerning, style editing/adding/removing. The formatting change may help draw the user's attention to the specific portion by rendering the specific portion inconsistent or distinct from other parts of the document, site, etc. Additionally or alternatively, information in the target document may be modified, for example, by adding, removing or editing relevant (or irrelevant) information, such as by scrolling to, altering or adding a link, adding an image, deleting surrounding aspect, adding new text, adding a popup or hover window, adding and/or executing JavaScript instructions and/or other computing instructions in other languages or methods. The information change may add links to relevant information known to the source page, to provide a method of returning to the source page, to add known links to the target page, etc.

[0033]   In one embodiment of the present invention, navigation by the browser directly to the specified query-relevant intra-document portion is implemented on the client side while an instruction to the client browser to navigate directly to the intra-document portion is provided by a source, e.g., a server or search engine, providing the clickable link. In particular, in one embodiment of the invention the client browser has installed therein an artificial named anchor module that may be implemented via a

toolbar such as the Google toolbar, a dynamic link library (DLL) or any other type of plug-in, a browser helper object (BHO), or any other suitable mechanism to implement the desired functionality in the browser. Note that the functionality need not be implemented using a module and that the browser, shell or Word document viewer, etc.

5   can implement the functionality natively without an add-on. The artificial named anchor module enables the browser to recognize when an URL includes an artificial named anchor. In one embodiment, the artificial named anchor module may be implemented to recognize as artificial any named anchor that begins with a preassigned, artificial named anchor prefix. For example, any set of suitable preassigned text characters may be

10   utilized as the preassigned anchor prefix. In one embodiment of the invention, the preassigned anchor prefix is relatively obscure so as to reduce the likelihood of conflicts with actual anchors in the destination webpage. Any other suitable mechanism for recognizing an artificial named anchor may be utilized. As an example, the preassigned text characters may appear anywhere within the name of the artificial anchor, e.g., as a

15   suffix or anywhere within the artificial anchor.

[0034]   In the example shown in **FIG. 5**, "__g_ " is used as the pre-assigned artificial named anchor prefix. As shown in the address bar 28 in **FIG. 5**, the URL the client browser received is http://www.waiter.com/roundtable#__g_570+N.+Shoreline. As is evident, the URL references an artificial named anchor "__g_570 N. Shoreline" although

20   the target webpage may or may not include such an anchor. Upon recognizing that the named anchor is artificial, the browser parses or strips the preassigned artificial prefix, e.g., __g_, from the artificial anchor and searches for the remaining artificial anchor text (e.g., "570 N. Shoreline") in the target webpage. If the remaining artificial anchor text is not located, the browser may display the webpage at the top. Alternatively, if the

25   remaining anchor text is located, the browser may navigate directly to the portion of the webpage containing the remaining anchor text and may also highlight the text. In the above example, the remaining anchor text is the same as the intra-document portion of the target webpage. It is noted the artificial anchor text may optionally be encoded such that the client browser may unencode the artificial anchor text to extract the artificial

30   anchor text before navigating directly to the intra-document portion of the target

webpage. In other implementations, different or no encoding mechanisms can be employed.

[0035]   As is evident, the client-side process may be implemented via the artificial named anchor module installed in the browser. **FIG. 6** is a flowchart of an illustrative client-side process 200 for recognizing and processing artificial named anchors and navigating directly to and highlighting the query-relevant portion of the webpage. The process 200 begins with the user entering a query via a search engine interface at block 202. At block 204, the search engine returns the search results typically with one or more snippets for each hyperlink. At block 206, the user selects from the search results and clicks on a selected hyperlink to navigate to a target or destination webpage to which the hyperlink refers. It is noted that for each given search result, the search results page may provide any number of hyperlinks, optionally with reference to an artificial anchor. It is noted that certain search results may not provide a hyperlink that references an artificial named anchor as may be the case where the target is a PDF document, for example. Currently, the browser only displays a PDF document at the top of the document although the browser may be similarly modified so as to be able to search and navigate within the PDF document. A similar mechanism may also be implemented for Word, Excel, and/or various other documents with segmentable and/or searchable pieces.

[0036]   At block 208, the browser determines if the selected hyperlink references a named anchor. If not, then the navigation proceeds as normal at block 210. Alternatively, if the selected hyperlink references a named anchor, then the browser determines if the named anchor is an artificial anchor at block 212. As noted above, the browser may determine that an anchor is artificial if the anchor begins with a preassigned artificial anchor prefix e.g., __g__. If the browser determines that the anchor is not artificial, then the navigation proceeds as normal at block 210. Alternatively, if the browser determines that the anchor is artificial, then the browser may parse and optionally unencode the artificial anchor at block 214. In other implementations, different or no encoding mechanisms can be employed. For example, the browser may strip the artificial anchor of the preassigned anchor prefix, e.g., __g__. The browser then loads the destination webpage and directly navigates to and may also highlight, such as by modifying the formatting of and/or information in the portion of the destination webpage

specified by the artificial anchor at block 216. In effect, the browser typically would find, highlight and scroll to the first instance of the portion of the destination webpage specified by the artificial anchor.

[0037] It is noted that in one embodiment of the present invention if the client web browser does not have artificial named anchor module installed, then the browser will default to ignoring the artificial named anchor as such an anchor would not exist in the actual target webpage. However, if the client web browser (or the client device in general) does have the appropriate artificial named anchor module installed, the artificial named anchor module will process artificial anchors according to the exemplary client-side process 200 described above. Thus, the artificial anchor serves as an instruction to the artificial named anchor module or to the client browser in general.

[0038] The client-side process 200 may allow any source (referral webpage or document) to utilize the artificial named anchor mechanism such that any client device with the artificial named anchor module installed would perform the artificial anchor recognition and processing process 200 to navigate directly to the portion of the target webpage referenced by the artificial anchor. Alternatively, the artificial named anchor module may additionally examine the source (referral webpage) of a hyperlink with an artificial named anchor and only process those whose source is authorized. For example, the client process may determine whether a source is authorized by sniffing an appropriate cookie or by use of headers, for example.

[0039] Artificial anchors are generally most useful for searching for text within the target webpage. However, other items to be displayed in the browser and/or otherwise defined by the target webpage may also be searched. Examples include image name, tag identification (ID), tag name, location in the document object model (DOM), HTML byte offset, a general offset or location definition within the search result document, etc. In addition to using artificial anchors to find, highlight and scroll to the referenced text or other portion of the target webpage, the client-side process may alternatively or additionally perform other functions. For example, the client-side process may be implemented to find and highlight occurrences of specific words within the target webpage, e.g., by using a different preassigned artificial anchor prefix such as "_gh_." Thus, multiple occurrences of the specific word(s) or phrase(s) being searched would be

highlighted. As another example, the client-side process may be implemented to find and scroll to a specified image based on, for example, the image name, within the target webpage, e.g., by using a different preassigned artificial anchor prefix such as "_gi_." In addition, the artificial anchor module of the client device may alternatively perform a

5   fuzzy (best effort) search rather than an absolute or strict search when searching for the specific portion referred to by the artificial anchor.

[0040]   While the artificial anchors are recognized and processed by the client-side process 200, in one embodiment the artificial anchors are generated by a server-side process although a client-side process may be similarly implemented. **FIG. 7** is a

10   flowchart of an illustrative search engine server-side process 220 for generating and appending an artificial anchor to the URL of the search result. In particular, at block 222, the search engine receives a query from the user. At block 224, the search engine performs the search and generates the search result. Each search result may include an URL or hyperlink referencing a target or destination webpage and one or more snippets

15   extracted from the target webpage to which the corresponding hyperlink refers. Any suitable search methodology may be employed in block 224.

[0041]   For each search result 226, the search engine may determine whether the search result hyperlink already includes or references an anchor at block228. Search engines typically do not return hyperlinks with anchors as anchors are generally not

20   indexed. If such is the case, block 228 may be eliminated. If the URL already includes an anchor, then the search engine may continue onto the next search result at 226. Alternatively, the search engine may strip the anchor from the URL at block 230.

[0042]   If the search result URL already includes an anchor and the search engine performs block 230 or if the search result URL does not include an anchor, then the

25   search engine generates a modified search result URL by appending an artificial named anchor to the search result URL at block 232. For example, the search engine may use a preassigned artificial anchor prefix such as "_g_" or any other mechanism to signal to the client device that the named anchor is artificially generated.

[0043]   As noted above, each search result may include a search result hyperlink

30   referencing the target webpage and one or more snippets extracted from the target webpage to which the hyperlink refers. When there are multiple snippets associated with

a given search result, the snippets are typically separated an ellipsis, such as a three-dot ellipsis. The search engine may transform each snippet into at least one active snippet in which each active snippet is a hyperlink with an artificial anchor that references the portion of the target webpage containing the corresponding snippet or portion of the

5   snippet. For example, each snippet may optionally be parsed into multiple snippet segments in which the search engine transforms each snippet segment into a separate hyperlink each with a corresponding artificial anchor. In one embodiment, the snippet may be segmented in accordance with punctuation marks that the snippet contains. In an alternative embodiment, the active snippet may link to an artificial anchor that simply

10   points to the search term or phrase, for example.

[0044]   After the search engine generates the hyperlink with the artificial anchor, the process 220 then continues with the next search result at 226. After all search results have been processed, the search engine server transmits the search results page to the client device at block 234.

15   [0045]   In one alternative embodiment, the search engine may also determine if the target webpage already contains a named anchor at or near the snippet, either before or instead of performing block 232 as shown in the flowchart of **FIG. 8** illustrating a modified portion of the process 220. As shown, after determining that the URL does not already include an anchor at block 228, the search engine may determine if the target

20   webpage already contains a named anchor at or near the snippet at block 240. Block 240 may be performed before or instead of block 232. If the target webpage already contains a named anchor at or near the snippet as determined in block 240, then the search engine utilizes that anchor by appending the anchor to the URL of the search result at block 242 and the process continues with the next search result.

25   [0046]   Process 220 may be implemented on the server side although it may be alternatively or additionally implemented on the client side such as via the client-side artificial anchor module or other suitable mechanism. When the process of generating artificial anchors is implemented on the client side and/or when the artificial anchor generation process is not implemented by the search engine utilized by the user, the

30   client-side process may examine each snippet in each search result and generate a hyperlink referencing an artificial named anchor.

[0047]   As is evident, the combination of generating artificial anchors and recognizing and processing artificial anchors improves the user's web browsing experience by allowing the user to navigate directly to a specific intra-document portion of the target document or webpage corresponding to the relevant snippet. The artificial anchors can be used with any suitable search results such as the Google's standard search, Geo-Search, Froogle search, etc.

[0048]   The use of artificial anchors is merely one embodiment. For example, the server may provide hints or instructions to the client browser by inserting hidden tags in the results page or attributes on the A tags. The installed browser module on the client side skims the DOM of each results page and extracts the relevant information from these hints or instructions. Thus, when the user navigates to a target webpage or document, the installed browser module applies the desired result of these instructions to achieve similar functions. Client web browsers that do no expect such instructions, i.e., without the appropriate module installed, for example, would simply ignore the instructions. In such an embodiment, the user would not see an artificial anchor appended to the URL in the address bar of the client browser.

[0049]   In one embodiment, to minimize the impact of transmitting extra information from the server that will only be ignored by the client device or more specifically the client browser if the client device is not installed with the artificial anchor module, the server may sniff the relevant cookie on the client device and determine if the artificial anchor module is installed. To facilitate such a process, installation of the artificial anchor module may cause the relevant cookie to be created and modified. If the artificial anchor module is not installed, then the server, e.g., search engine, is informed that the artificial anchor module is not installed when the query is submitted and the search engine may proceed with the search without generating references to artificial links, for example.

[0050]   In an alternative embodiment, the mechanism for automatically scrolling to a specific portion of a target page may be implemented on the server side. With server side implementation, a client side artificial anchor module may not be needed. For example, in response to the user clicking on a particular snippet line, the server returns a cached page directly scrolled to a portion of the target document corresponding to the selected

snipped. In one embodiment, the search query itself and the particular snippet line number are provided to the server when the user clicking on a particular snippet line. The server then regenerates the snippets, i.e., rerun the snippet generation algorithm, and return the cached page using, for example, JavaScript or other mechanism to

5 automatically scroll to the portion of the cached page that corresponds to the selected snippet line. As another example, each snippet for a given target page is a link to a portion corresponding to the snippet in a cached page. In particular, the link to the cached page may include the query as well as additional information, e.g., a hint such as in the form of an anchor, to appropriately position or scroll the cached page in the

10 browser using a named anchor. The cached page in turn may include named anchors each prefixing a portion of the cached page corresponding to the snippet. In general, with server side implementation, a server may proxy the target page or an intermediary, e.g., a web proxy or IBM's Web Intermediary (WBI), may implement the server side functionality.

15 **[0051]** In one embodiment, the system may be configured such that the browser may simultaneously display all the snippet lines, such as with the use of multiple frames each with at least one of the snippet lines visible within the browser. Alternatively, the server may be configured so as to return a cached page within one or more frames within a frame set with each frame scrolled to display one of the snippet lines. For example,

20 where a snippet generation engine returns three snippet lines, the browser may contain three frames within a frame set, with each frame displaying the same target or cached page and each scrolled to a different corresponding one of the snippet lines so as to simultaneously display all of the snippet lines generated by the snippet engine.

[0052] **FIG. 9** illustrates an exemplary networked system 100 in which systems and

25 methods described herein may be implemented. The networked system 100 may include client devices 102 in communication with servers 104 and 106 via a network 108. The network 108 may be a local area network (LAN), a wide area network (WAN), a telephone network, such as the Public Switched Telephone Network (PSTN), an intranet, the Internet, or any suitable combination of networks. For purposes of clarity, two client

30 devices 102 and three servers 104 and 106 are illustrated as connected to the network 140. However, any suitable number of client devices 102 and servers 104, 106 may be

connected via the network 140. In addition, a given client device may perform the functions of a server and a server may perform the functions of a client device. The client devices 102 may include devices, such mainframes, minicomputers, personal computers, laptops, personal digital assistants, or the like, capable of connecting to the network 108. The client devices 102 may transmit data over the network 108 and/or receive data from the network 108 via a wired (e.g., copper, optical, etc.) and/or wireless connection.

[0053]   **FIG. 10** illustrates an exemplary client device 102 suitable for implementation in the networked system 100 of **FIG. 9**. The client device 102 may include a bus 120, a processor 122, a main memory 124, a read only memory (ROM) 126, a storage device 128, an input device 130, an output device 132, and a communication interface 134. The bus 120 may include one or more conventional buses that permit communication among the components of the client device 102. The processor 122 may include any type of conventional processor or microprocessor that interprets and executes instructions. The main memory 124 may include a random access memory (RAM) or another type of dynamic storage device that stores information and instructions for execution by the processor 122. The ROM 126 may include a conventional ROM device or another type of static storage device that stores static information and instructions for use by the processor 122. The storage device 128 may include a magnetic and/or optical recording medium, for example, and its corresponding drive.

[0054]   The input device 130 may include one or more conventional mechanisms that permit a user to input information to the client device 102 such as a keyboard, a mouse, a pen, voice recognition and/or biometric mechanisms, etc. The output device 132 may include one or more conventional mechanisms that output information to the user, including a display, a printer, a speaker, etc. The communication interface 134 may include any transceiver-like mechanism that enables the client device 102 to communicate with other devices and/or systems. For example, the communication interface 134 may include mechanisms for communicating with another device or system via a network, such as network 108.

[0055]   The client devices 102 perform certain searching-related operations such as those described above. The client devices 102 may perform these operations in response

to the processor 122 executing software instructions contained in a computer-readable medium, such as memory 124. A computer-readable medium may be defined as one or more memory devices and/or carrier waves. The software instructions may be read into memory 124 from another computer-readable medium such as the data storage device

5   128 or from another device via the communication interface 134. The software instructions contained in memory 124 causes processor 122 to perform search-related activities described below. Alternatively, hardwired circuitry may be used in place of or in combination with software instructions to implement search-related processes described herein. Thus, the present invention is not limited to any specific combination of

10   hardware circuitry and software.

[0056]   The servers 104 and 106 may include one or more types of computer systems, such as a mainframe, minicomputer, or personal computer capable of connecting to the network 108 to enable servers 104, 106 to communicate with the client devices 102. In alternative implementations, the servers 104, 106 may include mechanisms for directly

15   connecting to one or more client devices 102. The servers 104, 106 may transmit data over the network 108 or receive data from the network 108 via a wired or wireless connection.

[0057]   The servers may be configured in a manner similar to that described above in reference to **FIG. 10** for client device 102. In one implementation, the server 106 may

20   include a search engine 110 usable by the client devices 102. The servers 104 may store documents (e.g., web pages) accessible by the client devices 102.

[0058]   While the exemplary embodiments of the present invention are described and illustrated herein, it will be appreciated that they are merely illustrative and that modifications can be made to these embodiments without departing from the spirit and

25   scope of the invention. Thus, the scope of the invention is intended to be defined only in terms of the following claims as may be amended, with each claim being expressly incorporated into this Description of Specific Embodiments as an embodiment of the invention.